IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

First Named Applicant: Cha	)	Art Unit: 2177
Serial No.: 09/512,949	)	Examiner: Pannala
Filed: February 25, 2000	)	**AM9-99-0217**
For: **INDEXING SYSTEM AND METHOD FOR NEAREST NEIGHBOR SEARCHES IN HIGH DIMENSIONAL DATA SPACES**	)	June 25, 2002 750 B STREET, Suite 3120 San Diego, CA 92101

APPEAL BRIEF

Commissioner of Patents and Trademarks
Washington, DC 20231

Dear Sir:

This appeal brief is submitted under 35 U.S.C. §134. This appeal is further to Appellant's Notice of Appeal filed herewith.

Table of Contents

(1) Real Party in Interest

The real party in interest is IBM Corp.

(2) Related Appeals/Interferences

No other appeals or interferences exist which relate to the present application or appeal.

(3) Status of Claims

Claims 1-24 are pending and twice rejected.

(4) Status of Amendments

No amendments are outstanding.

(5) Summary of Invention

As set forth in Claim 1, the invention is a computer programmed to query for data. For data vectors in a data space, respective approximations are generated in polar coordinates, and then based on the approximations, "k" nearest neighbors to the query are returned.

(6) Issues

(a) Whether the claims are unpatentable under 35 U.S.C. §103 as being unpatentable over Fayyad et al. in view of Apple Computer.

(b) Whether the claims are unpatentable under 35 U.S.C. §112, first paragraph for allegedly lacking an enabling disclosure.

(7) Grouping of Claims

The Claims stand together.

(8a) Argument

Claims 1-4, 8-12, 15-18, and 22-24, all of which require using polar coordinates, have been rejected under 35 U.S.C. §103 as being unpatentable over Fayyad et al., which nowhere mentions the word "polar" or appears to suggest the use of anything other than the Cartesian coordinates disclosed in, e.g., equations 3 and 4 in column 8, in view of Apple Computer's teaching of how to convert Cartesian coordinates to polar. Also, Claims 5-7, 13, 14, and 19-21 have been rejected under 35 U.S.C. §103 as being unpatentable over Fayyad et al. in view of Staats, another patent in which the word "polar" nowhere appears.

Appellant does not dispute that, as a general abstract proposition, it is possible to convert from one coordinate system to another. But what is missing in the rejection is an explanation of why the prior art motivates one to change Fayyad et al.'s Cartesian system to polar. All the examiner offers by way of explanation is that one would have been motivated to do so because Apple Computer provides an easy way to convert. But the ease or difficulty of a proposed modification is irrelevant if the modification itself is not suggested in the prior art. Apple Computer, of course, has nothing at all to do with querying, so it cannot suggest modifying a querying system such as Fayyad et al.'s to use polar coordinates, and Fayyad et al. is entirely directed to a Cartesian system. It is presumed that Fayyad et al.'s inventors were aware of the

existence of polar coordinates, yet nowhere did they choose to mention polar coordinates. They considered only Cartesian coordinates, and for good reason.

Specifically, the entire thrust of Fayyad et al.'s querying system is cast in column 8 in terms of "Euclidean distance". None of the equations on which Fayyad et al. is based would work in polar coordinates. Indeed, no proffer of why a reasonable expectation of success exists in converting Fayyad et al.'s invention to polar has been made, contrary to the requirements of MPEP §2142. This is perhaps not surprising, because Fayyad et al. would appear to require wholesale modification if it were to depart from its Cartesian coordinate scheme.

As it is, simply converting Fayyad et al.'s Cartesian system to polar would destroy the efficacy of its equations to execute nearest neighbor searches, thereby rendering the proposed modification deficient under MPEP §2143.01 (citing In re Gordon). For example, consider the disclosure in Fayyad et al., col. 8, lines 14-25:

> "Use of a Euclidean weighted distance measure does not change the results other than a pre-scaling of the input space...[t]he use of a weighting factor allows certain dimensions of the *n* attributes to be favored or emphasized."

Thus, it is important to the Fayyad et al. invention that a Euclidean weighting factor (and concomitantly Cartesian coordiantes) be used to discriminate among attributes, as taught by Fayyad et al. But, how this desired feature might be achieved were Fayyad et al. converted to polar coordinates is anyone's guess. Certainly, if anything Fayyad et al.'s emphasis on the advantages of its disclosed Cartesian coordinate system, coupled with its absolute silence on any other coordinate system, would hardly motivate one to

modify Fayyad et al. to use polar coordinates. Accordingly, the rejection, which simply observes that one can convert from Cartesian to polar coordinates without identifying any prior art reason to do so in the context of data querying, fails to establish a prima facie case of obviousness, and should be reversed.

(8b) Argument

The claims have been rejected under 35 U.S.C. §112, first paragraph for allegedly lacking an enabling disclosure. The examiner has alleged that the "dimensionality d" of Claims 3, 8, 11, and 13 lack enabling disclosure, as well as "dmin1 < K-NNdist(q)" of Claim 13 and "approximations" of Claims 1, 8, and 15 for being based on "k-NNdist".

The examiner bases his rejection on the allegation that it is unclear whether NNdist is a single variable, or whether it is the product of N times Ndist where dist is a power of N, and further that k-NNdist may represent a subtraction. Furthermore, per the examiner "other interpretations can be considered, but the above are examples which would cause undue experimentation".

Well, other interpretations can be considered and undue experimentation ensue, provided one has not read the specification. If one actually reads the specification, however, the alleged opaqueness evaporates.

Specifically, consider page 4, lines 6 and 7, wherein it is disclosed that $k\text{-}NN^{dist}$ (**q**) is the k^{th} largest distance between the query vector **q** and nearest neighbor vectors **p** encountered so far. This clearly defines what the value is. It is a distance, not a subtraction or a product. "$k\text{-}NN^{dist}$" is thus simply the notation that the inventors used to represent the clearly disclosed and enabled distance. Moreover, on page 9, lines 8 and 9, it is further disclosed that the distance $k\text{-}NN^{dist}(q)$ is initialized at an appropriate large value. It has not

been shown or alleged that selecting such an initial value would require undue experimentation, much less why.

At page 10, lines 1-3, it is disclosed that subsequent to initialization, k-NN^{dist}(**q**) can be recomputed by setting it equal to the k^{th}-largest distance k-NN^{dist}(**q**) corresponding to the k^{th} vector **p** in the answer set. It has not been alleged, much less explained, why this straightforward and easy to comprehend step would require undue experimentation. Accordingly, the rejection of the independent claims has been overcome.

Turning to the allegation that the "dimensionality of d" in Claims 3, 8, 11, and 13 and "d_{minl} < k-NNdist(q)" in Claim 13 lack enablement, attention is directed to page 7, line 10, wherein it is stated that "d" is the dimensionality of the database 20. The examiner states that "the broadest reasonable interpretation of "d" could be constructed as any integer number" (sic), and then alleges that since the specification specifically discloses a preferred embodiment wherein d=2 or 3, the claims, which do not set a limit on "d", are not enabled.

It is axiomatic that the claims are not limited to what is disclosed. In this case, dimensionalities of 2 and 3 are used as mere examples; the fact that no upper bound on dimensionality is set does not mean that the broad interpretation of "d" is not enabled, since the skilled artisan knows that databases can have very high dimensionalities (see the first line of the present Background) and that the principles of the invention apply to any dimensionality. In fact, page 7, lines 10-14 explode the basis for this rejection:

> "***For illustration purposes***, Figure 3 shows a two-dimensional data space that has been divided into plural cells 26, while Figure 4 illustrates a single three dimensional cell 28. ***The use of two and three dimensions in Figures 2 and 3 is for simplicity of disclosure only, it being understood that the principles set forth herein apply to any high dimensional data spaces.***"

This leaves "d_{min1} < k-NNdist(q)" in Claim 13. The second term has been discussed above, and is fully enabled. The examiner concedes that d_{min1} is stated in the summary, but complains that "it is not supported or explained more about it (sic) in the specification. The applicant states that d_{min1} is the first approximation, but the lack of details in the specification is enabling (sic) and leading to undue breadth (sic). It also raises other questions like, what is the value/range for d_{min}?"

The examiner has failed to recognize right answer when told. Specifically, Appellant has pointed out that on page 4, line 5, it is disclosed that d_{min1} is a lower bound on an approximation, with the numeral "1" designating that the bound is related to a claimed "first" approximation for clarity of claiming. On page 8, second full paragraph, d_{min}, which is also referred to as a minimum distance, is equal to $[\,|\mathbf{p}|^2 + |\mathbf{q}|^2 - 2\,|\mathbf{p}|\,|\mathbf{q}|\cos(\theta_1 - \theta_2)]^{1/2}$, wherein the angle θ_1 is the angle between the cell diagonal and the data vector **p** and the angle θ_2 is the angle between the cell diagonal and the query vector **q**. In other words, despite the examiner's bewilderment a detailed equation with defined terms has been given for d_{min}, fully enabling this term. The rejection accordingly should be reversed.

Respectfully submitted,

John L. Rogitz
Registration No. 33,549
Attorney of Record
750 B Street, Suite 3120
San Diego, CA 92101
Telephone: (619) 338-8075

JLR:jg

APPENDIX A

1. A computer programmed to undertake method acts for querying for data using a query, the method acts undertaken by the computer including:

for at least some data vectors in a data space, generating respective approximations in polar coordinates; and

based on the approximations, returning "k" nearest neighbors to the query.

2. The computer of Claim 1, wherein the method acts further comprise:

dividing the data space into plural cells; and

representing at least one data point in at least one cell in polar coordinates with respect to the at least one cell.

3. The computer of Claim 2, wherein the data space has "d" dimensions and the method acts further comprise:

determining a number of "b" bits to be assigned to each cell; and

dividing the data space into 2^{bd} cells.

4. The computer of Claim 1, wherein each approximation defines a lower bound d_{min}, and the method acts further comprise:

generating a candidate set of approximations based at least on the lower bounds d_{min} of the approximations.

5. The computer of Claim 4, wherein the query can be represented by a query vector **q**, and the method acts further comprise:

adding a first approximation having a first lower bound d_{min1} to the candidate set if d_{min1} < k-NN^{dist} (q), wherein k-NN^{dist} (q) is the k^{th} largest distance between the query vector **q** and nearest neighbor vectors **p**.

6. The computer of Claim 5, wherein the method acts further comprise using the candidate set to return "k" nearest neighbors vectors **p** to the query vector **q**.

7. The computer of Claim 6, wherein not all vectors **p** corresponding to approximations in the candidate set are examined to return the "k" nearest neighbors.

8. A computer program product including a program of instructions having:

computer readable code means for generating approximations including local polar coordinates of at least some data vectors **p** in at least one data set having a dimensionality of "d", the local polar coordinates being independent of "d"; and

computer readable code means for using the approximations to return "k" nearest neighbors to a query.

9. The computer program product of Claim 8, wherein the means for generating generates respective approximations of data vectors **p** in local polar coordinates.

10. The computer program product of Claim 9, further comprising:

computer readable code means for dividing the data space into plural cells; and

computer readable code means for representing each approximation in polar coordinates with respect to one of the cells.

11. The computer program product of Claim 10, wherein the data space has "d" dimensions, further comprising:

computer readable code means for determining a number of "b" bits to be assigned to each cell; and

computer readable code means for dividing the data space into 2^{bd} cells.

12. The computer program product of Claim 9, wherein each approximation defines a lower bound d_{min} and an upper bound d_{max}, and the product further comprises:

computer readable code means for generating a candidate set of approximations based at least on the lower bounds d_{min} and upper bounds d_{max} of the approximations.

13. The computer program product of Claim 12, further comprising:

computer readable code means for adding a first approximation having a first lower bound d_{min1} to the candidate set if d_{min1} < k-NN^{dist} (**q**), wherein k-NN^{dist} (**q**) is the k^{th} largest distance between the query vector **q** and nearest neighbor vectors **p** associated with approximations in the candidate set.

14. The computer program product of Claim 13, further comprising computer readable code means for using the candidate set to return "k" nearest neighbors vectors **p** to the query vector **q**.

15. A computer-implemented method for finding, in a data space, "k" closest data vectors **p** to a query vector **q**, comprising:

rendering approximations of at least some of the data vectors **p** using local polar coordinates;

filtering the approximations; and

after filtering, returning the "k" closest data vectors **p**.

16. The method of Claim 15, further comprising:

dividing the data space into plural cells; and

representing each approximation in polar coordinates with respect to one of the cells.

17. The method of Claim 16, wherein the data space has "d" dimensions and the method further comprises:

determining a number of "b" bits to be assigned to each cell; and

dividing the data space into 2^{bd} cells.

18. The method of Claim 15, wherein each approximation defines a lower bound d_{min}, and the method further comprises:

generating a candidate set of approximations based at least on the lower bounds d_{min} of the approximations.

19. The method of Claim 18, further comprising:

adding a first approximation having a first lower bound d_{min1} to the candidate set if d_{min1} < k-NN^{dist} (**q**), wherein k-NN^{dist} (**q**) is the k^{th} largest distance between the query vector **q** and nearest neighbor vectors **p** associated with approximations in the candidate set.

20. The method of Claim 19, further comprising using the candidate set to return "k" nearest neighbors vectors **p** to the query vector **q**.

21. The method of Claim 20, wherein not all data vectors **p** corresponding to approximations in the candidate set are examined to return the "k" nearest neighbors vectors **p**.

22. The computer of Claim 4, wherein each approximation defines an upper bound d_{max}, and the method acts further comprise:

generating a candidate set of approximations based at least on the upper bounds d_{max} of the approximations.

23. The computer program product of Claim 12, wherein each approximation defines an upper bound d_{max}, and the product further comprises:

computer readable code means for generating a candidate set of approximations based at least on the upper bounds d_{max} of the approximations.

24. The computer of Claim 1, wherein each approximation defines an upper bound d_{max}, and the method acts further comprise:

generating a candidate set of approximations based at least on the upper bounds d_{max} of the approximations.